

# RELATÓRIO

## ucBusca

Trabalho realizado por:

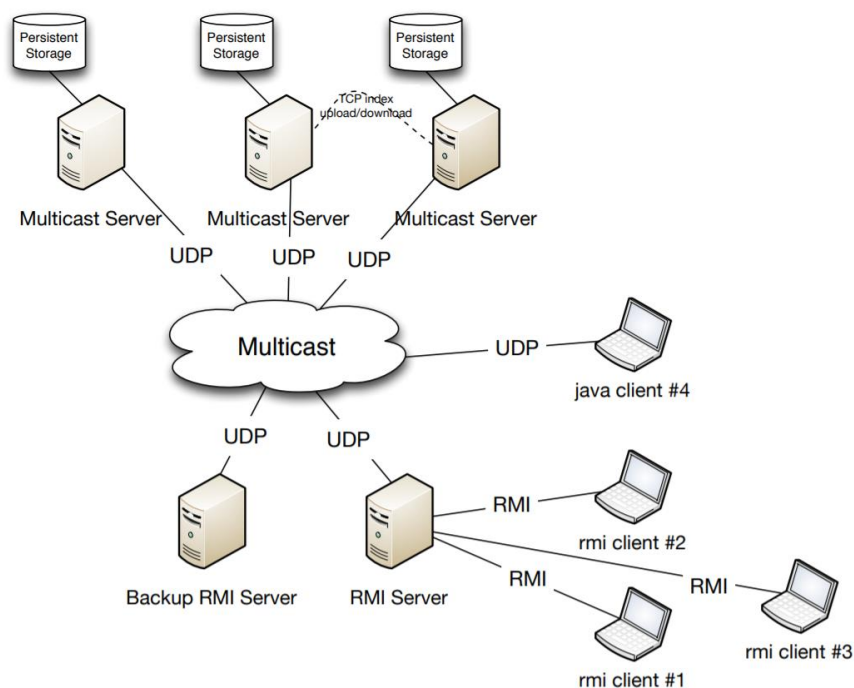
Luís Freitas, 2017264549

João Monteiro, 2016244006

## ARQUITETURA

A arquitetura do no nosso projeto é inteiramente baseada no enunciado, onde o RMIClient invoca os métodos do RMIServer (e vice-versa). Onde existirá um backup do RMIServer que assumirá lugar caso o que até então se encontrar ativo falhe.

O RMIServer vai ainda processar “os pedidos” do RMIClient e enviar uma mensagem ao Multicast, para que este possa aceder ao dados necessários e responder ao RMIServer para estedar o return correto nos métodos invocados pelo RMIClient.



## PROTOCOLO MULTICAST

O Multicast e o RMI Server estão conectados através de MulticastSockets. Uma MulticastSocket envia e recebe pacotes utilizando o protocolo UDP, tendo a capacidade de se juntar a grupos de outros endereços multicast na internet. O endereço IP "224.0.1.0" e a porta "4321" caracteriza o servidor Multicast e é a partir destes que se ligam os servidores RMI. Cada vez que uma função do servidor RMI é chamada pelo Cliente, é criada uma MulticastSocket que se conecta ao servidor principal, invocando o método `joinGroup(InetAddress.getByName(MULTICAST_ADDRESS))` e de seguida, envia uma mensagem com a informação e um comando específico, requerindo a resposta à ação do cliente.

## COMUNICAÇÃO RMI CLIENTE - SERVIDOR

O RMI Client, tal como o nome "RMI" indica, irá invocar os métodos presentes no RMI Server para que este com a informação que o RMI Client passa como input dos métodos remotamente invocados, possa construir a mensagem que será posteriormente enviada ao Multicast.

Uma vez que o RMI Server recebeu a mensagem do Multicast interpreta os resultados e envia para o cliente a informação que este deve apresentar.

O RMI Server também irá invocar métodos do RMI Client mas apenas nos casos de o RMI Server receber uma mensagem do multicast a informar que o cliente deve receber uma notificação, e aí o server irá chamar a função que permite ao cliente receber a notificação em tempo real.

## DISTRIBUIÇÃO DE TAREFAS

João Monteiro - Servidor Multicast

Luís Freitas - Servidor e Cliente RMI

### Testes Realizados

TESTES REALIZADOS	PAS S	FAI L
Registar novo utilizador	X	
Acesso protegido com password (exceto pesquisas)	X	
Indexar novo URL introduzido por administrador	X	
Indexar iterativamente ou recursivamente todos os URLs encontrados	X	
Pesquisar páginas que contenham um conjunto de palavras	X	
Resultados ordenados por número de ligações para cada página		X
Consultar lista de páginas com ligações para uma página específica	X	
Consultar lista de pesquisas feitas pelo próprio utilizador	X	
Dar privilégios de administrador a um utilizador	X	
Página de administração atualizada em tempo real		X
Notificação imediata de privilégios de administrador (online users)	X	
Entrega posterior de notificações (offline users)	X	